2018-04

Automatic Generation of System Plans for Autonomous Unmanned Systems

Berzins, Valdis

Monterey, California: Naval Postgraduate School

https://hdl.handle.net/10945/64377

MONTEREY, CALIFORNIA

AUTOMATIC GENERATION OF SYSTEM PLANS FOR AUTONOMOUS UNMANNED SYSTEMS

Report Type: Interim Report
Period of Performance: 01/01/2018-12/31/2018

Project PI: Valdis Berzins, Professor of Computer Science

Prepared for:
Topic Sponsor: ASN(RDA) - Research, Development, and Acquisition
Research Sponsor Organization: Space and Naval Warfare Systems Command (SPAWAR)
Research POC Name: CAPT Kurt Rothenhaus
Research POC Contact Information: kurt.rothenhaus@navy.mil, 619-524-7909

EXECUTIVE SUMMARY

Project Summary

The objective of the study is to leverage automatic programming in support of Systems Center Pacific (SSC-P) strategies for developing unmanned systems autonomy software. To apply artificial intelligence (AI) planning algorithms to this problem, research on suitable representations for mission plans and system plans is needed, along with principles for automatically deriving system plans from mission plans.

The study also seeks to improve man-machine teaming for controlling unmanned/autonomous systems that perform Navy missions. Mission plans should be described at a conceptual level natural for the human operators of autonomous systems and suitable for enabling one operator to control a team of multiple autonomous systems. System plans must be sufficiently clear and detailed to be carried out by fully automated systems. Specific research objectives are to investigate the following questions:

- What kinds of mission models are needed to support mission planning for teams of autonomous systems?
- How can we formulate the distributed adaptive planning general case to allow rigorous research in the area?
- How can autonomous systems plans be represented to enable automatic system plan generation from mission plans?

Keywords: *unmanned systems, autonomy, unmanned strategy, model driven development, automatic programming, planning*

Background

Prior software engineering research gave us breakthroughs in technology that improved the development of computer software and benefited real-time embedded systems. The automatic programming approach pioneered by the Naval Postgraduate School (Luqi & Berzins, Rapid Prototyping of Real-Time Systems, 1988, Luqi, Berzins, & Yeh, A Prototyping Language for Real Time Software, 1988, Luqi & Berzins, Execution of a High Level Real-Time Language, 1988, and Luqi & Berzins, Semantics of a Real-TIme Language , 1988) utilized formal representations of system requirements, and in particular timing relationships among operations. Teams of autonomous systems, such as the Unites States Navy (USN) is envisioning deploying in the future, share many of the characteristics that made these formal mathematical methods, scheduling, and code generation so useful. Additional factors include uncertainty and a larger set of possible actions and states. Mathematical formulations of the problem with a rigorous approach must be used. There are set theoretic, statistical and information theoretic aspects to be integrated for the application.

Previous work on handling uncertainty in plan generation for autonomous systems has focused on learning control in unknown environments and dealing with sensor uncertainty (Busoniu & Tamas, 2015). Most successful previous work on automatic program generation has focused on translation from goal models expressed using domain-specific higher-level languages, such as, “A Prototyping Language for Real Time Software” by Luqi, Berzins, & Yeh, “Execution of a High Level Real-Time Language” by Luqi & Berzins,“The Realizable Benefits of a Language Prototyping Language” by Herndon & Berzins, and “Generative Programming: Methods, Tools, and Applications” by Czarnecki & Eisenecker. Other previous work includes “Partial Evaluation and Automatic Program Generation” by Jones, Gomard, & Sestoft, “A New Architecture for Transformation-Based Generators” by Biggerstaff, and “Design Wizards and Visual Programming Environments for GenVoca Generators” by Batory, Gang, Robertson, & Tao.

Findings and Conclusions

This project is ongoing. We analyzed the references cited in this proposal as well as additional sources of information on control of multiple unmanned vehicles (UxVs), automatic software generation, and representation of stochastic scenarios, both published and from consultations with appropriate subject matter experts.

We also identified and analyzed relevant previous work on control of multiple UxVs, UxV mission modeling and planning, automatic software generation, and representation of stochastic scenarios. Then we consulted subject matter experts on background material and obtained feedback on the appropriateness of proposed frameworks and characterization of uncertain factors that will affect plans, identified relevant sets of mission states and possible UxV actions, and defined/modeled concepts related to vehicle control. We formulated initial mission and plan models, obtained feedback, and used an iterative refinement process to achieve adequacy. Thesis students will carry out case studies for validating adequacy of proposed representations of mission plans and vehicle plans representations for usability in realistic situations.

We interacted with subject matter experts from Lockheed Martin on representation of mission plans for the anti-submarine warfare continuous trail unmanned vessel (ACTUV) Sea Hunter. Associated analyses and background studies produced these conference publications (Berzins, Putting Teeth into Open Architectures: Infrastructure for Reducing the Need for Retesting, 2018, Berzins, Automated Methods for Cyber Test and Evaluation, 2018, and Berzins & Hernandez, Risk-Based Testing for Drones, 2018).

Recommendations for Further Research

If done properly, man-machine teaming can result in better performance than either isolated world-class human experts or isolated world-class software. Success in this direction depends on automated support for mapping high-level mission plans formulated in terms that are natural for human commanders into more detailed system-level plans that can be carried out autonomously by UxVs. Future research is recommended to explore particular UxV mission areas in greater depth, determine alternatives for how UxV commanders could interact with the swarms they direct, develop plan generation schemata and associated plan generation software, and then conduct exercises to determine which models work better in which situations.

References

Batory, D., Gang, C., Robertson, E., & Tao, W. (2000, May). Design Wizards and Visual Programming Environments for GenVoca Generators. IEEE Transactions on Software Engineering, 26(5), 441-452.

Berzins, V. (2018). Automated Methods for Cyber Test and Evaluation. Proceedings of the 15th Annual Acquisition Research Symposium, 2, pp. 280-288. Monterey, CA.

Berzins, V. (2018). Putting Teeth into Open Architectures: Infrastructure for Reducing the Need for Retesting. Collaborative Open Systems Architecture Workshop 4. Monterey, CA.

Berzins, V., & Hernandez, A. (2018). Risk-Based Testing for Drones. 86th MORS Symposium. Monterey, CA.

Biggerstaff, T. (2004, December). A New Architecture for Transformation-Based Generators. IEEE Transactions of Software Engineering, 30(12), 1036-1054.

Busoniu, L., & Tamas, L. (2015). Handling Uncertainty and Networked Structure in Robot Control. Springer International Publishing.

Czarnecki, K., & Eisenecker, U. (2000). Generative Programming: Methods, Tools, and Applications. Pearson.

Herndon, R., & Berzins, V. (1988). The Realizable Benefits of a Language Prototyping Language. IEEE Transactions on Software Engineering, 803-809.

Jones, N. D., Gomard, C. K., & Sestoft, P. (1999). Partial Evaluation and Automatic Program Generation. Prentice Hall International.

Luqi, & Berzins, V. (1988). Execution of a High Level Real-Time Language. Proceedings of the 9th IEEE Real-Time Systems Symposium (pp. 69-78). Huntsville, AL: IEEE.

Luqi, & Berzins, V. (1988). Rapid Prototyping of Real-Time Systems. IEEE Software, 25-38.

Luqi, & Berzins, V. (1988). Semantics of a Real-TIme Language . Proceedings of the 9th IEEE Real-Time Systems Symposium (pp. 106-110). Huntsville, AL: IEEE.

Luqi, Berzins, V., & Yeh, R. (1988). A Prototyping Language for Real Time Software. IEEE Transactions of Software Engineering, 1409-1423.

Acronyms

ACTUV - ASW continuous trail unmanned vessel
AI - artificial intelligence
ASW - anti-submarine warfare
SPAWAR - Space and Naval Warfare Systems Command
SSCP - Systems Center Pacific
USN - United States Navy
UxV - unmanned vehicle